DOCUMENT RESUME

ED 243 474 IR 011 100

AUTHOR Brown, Geraldine E.; Clark, Brenda
TITLE Prescription for Computing Success: Knowledge and Self-Awareness.
PUB DATE [84]
NOTE 43p.
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Check Lists; Computer Assisted Instruction; *Computer Literacy; Computer Managed Instruction; Computers; Computer Software; *Decision Making; Disabilities; Educational Administration; *Independent Study; Problems; *Programing; Self Evaluation (Individuals); *Skill Development; *Special Education Teachers; Telecommunications

ABSTRACT

Intended for use by educators of handicapped individuals, this publication uses the form of an allegory to describe the process of learning to use computers, and provides guidelines for novices who want to utilize computers in the classroom. The need for self-awareness, initial planning, and acquisition of basic knowledge about computers is emphasized, and information is provided on user problems, educational software, programing, the process of teaching students to use computers, computerized instructional management, administrative use of computers, and use of telecommunications technology to link computers. For each topic covered, a definition, state-of-the-art information, and a list of problems--based in some cases on surveys of special educators--are presented. Also provided are descriptions of a visit to a computer store and a computer literature reading program, as well as information on the decision making process involved in the search for a suitable course on computer terminology and programing. A list of the personal and situational assessment factors involved in this decision making process and a sample chart illustrating rating of the important factors are presented. A 29-item bibliography is included. (ESR)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED243474

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
- Minor changes have been made to improve reproduction quality.

- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

Prescription for Computing Success: Knowledge and Self-Awareness

Geraldine E. Brown

And

Brenda Clark

George Peabody College for Teachers of Vanderbilt University
Media Technology Center
Box 45
Nashville, Tennessee
37203

IR011100

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Geraldine E. Brown

Brenda Clark

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Prescription for Computing Success: Knowledge and Self-Awareness

Learning to use computers can be a lonely, frustrating, time-consuming and expensive process. You can reduce these negative side-effects by practicing the forgotten art of long-range planning, while heeding the ancient adage: Know thyself. Planning entails decision-making. Good decision-making requires information on all the constraints and contingencies, which can effect a plan. Gaining access to accurate, up-to-date information in the world of computers, is a formiddable task. The field of computer technology is growing and changing daily (Williams, 1983). Thus, a fact today, may be incorrect, or out-dated information tomorrow. In addition, there are factors in your life, which will either constrain or facilitate the process of learning about computers. Such factors include: your amount of free-time, your budget, your feelings about machines, your reactions to success and failure, and your beliefs about learning in general. Thus, your success in learning to use computers is directly related to both, your awareness of the computer field, and awareness of self.

You are responsible for determining the role of the computer in your classroom; an important decision. Computers can be used in a variety of ways. Your first task, after you have increased your awareness, will be to answer the question, How can I use

this tool, the computer, to assist me in my work?

Take time to think. Would you like to speed up your paperwork? Is there something extra you would do, if you had more time, like writing notes home to parents? Could your students benefit from more individual instruction?

Your decision will determine the strength of your motivation, as you learn. However, you will change your mind, as you learn about computers. You will discover other tasks, it can do, to make your work easier.

This paper is based on two assumptions: first, that you are interested in computers, and second, that you have defined your role as a professional. In our opinion, your role, as a special educator, includes serving as an adult model, while preparing handicapped individuals to cope with an uncertain future. What does this mean?

Our society is in the midst of a transition, from an industrial state to a technological state (Toffler, 1980). Forecasts or predictions of the future are, at best, tenable. We don't know what kinds of jobs will exist, or how individuals will gain access to information they need. Will all information be computerized? Where will people work? What type of skills will our handicapped youth need to compete in the job market? What will workers do when they have been replaced by robots?

In his book, <u>The Third Wave</u>, Alvin Toffler predicts that the computer will change the business world, the nature of work, and is certain to effect the structure of the family and

education (1980). One fact is certain, computers will have a profound influence on all workers (Toffler, 1983).

The status, and standard of living of individuals in our society, is based on their success in the world of work. Will computer skills just broaden the range of available opportunities (Wheatley, 1983)? Or, will employers use computer training as a starting point, for screening new employees (Harvey, 1982)?

It is apparent that time is past for pleas to special educators to become computer literate (Stallard, 1982). It is time for these machine to be used in special education classrooms. It is also time to be realistic. Computers are not user-friendly, nor is it easy to learn to use them. Friendly connotes smiling, and understanding. A machine that erases 17,000 words on your disc because you hit the wrong key, is neither smart, understanding, or friendly. Is it smiling as it sends electronic signals to a printer, that has printed 500 words on the same line, because the paper is jammed? Wouldn't a friendly computer send a signal to the printer to stop?

Are we afraid of machines which can process information faster than we can? Are we reluctant to enter a land of technology, where researchers are teaching computers to imitate our verbal behaviors,and physical actions(Brooks,1983; Lenat, & Brown,1983)? Never fear, the fifth-generation computers, the intelligent machines, human clones, are still a vision of the future (Feigenbaum, & McCorduck, 1983).

Oh, you say you are waiting for computers that talk, and

respond to the verbal commands of the user, because your students have reading problems, and can't type or spell. Well, by the time those sophisticated machines are developed, and marketed at reasonable prices, our elementary-age students will be seeking employment. Our students are losing valuable time, as we forestall opening the door to the land of computers.

Perhaps if you are forewarned of the problems you will encounter, you can equip yourself, at least mentally, for your journey. Can some obstacles be avoided, by finding alternate routes? How are computers used by other special educators, and how do you learn to speak techno-babble (Lopez, 1983), the languge of the inhabitants of this land?

General uses for computers

Let's open the door to this land of foreign machines, and strange-speaking people, and find out what directions other special educators have gone. We can use our binoculars for our search. Let's peek in...ugh, the door is stuck. There... hm-m-m...not at all what we expected. A vast wilderness, with a series of mountain ranges in the distance. Strange...although the sky is blue, only the tops of the mountains are shimmering in the sunlight.

Etched on each mountainside are interwoven, snake-like grooves. These trails must lead to the top. We need to find someone, who knows this territory to help us. As we peer behind

the door, we notice a paved road leading to an old castle. Maybe someone is there, who can help us.

As we begin walking towards the castle, a brown furry creature with spindly legs runs across our path. "Wait," we shout, " We need some help."

He walks slowly back to us, eyes to the ground. "Where are all the computer users? Why does the sun shine at the top? How do we find the special educators?"

As he shuffles his feet anxiously, he speaks rapidly in a very low voice. " The users are at hostels throughout the land, reading, or working with computers. They are all working their way to the summits at the top of various mountains, where the sun shines on those who finally arrive."

"It should be easy to spot the well-trodden paths of the special teachers," he continues. "The trashcans along the way, are full of papers, with funny titles, like I.E.P., and data sheet. I have watched them rushing out of hostels with discs in hand, wildly looking for the nearest can. I am not fond of these people. They climb too many mountains, and pass too many others on the way up. Not only do they learn rapidly, even over on Programming Mountain, but they want to make things easier for everyone. Can you believe they had the audacity to mark all obstacles for novice travelers?"

"Never in my fifteen years of working on Expert Systems Mountain, have I seen anything like those special teachers!"

"Wait, don't go yet", we plead. " What are the names of the

mountains, where you have seen them? How much does it cost to stay in a hostel? And, what is that old castle up ahead?"

As he begins to shuffle his feet even more rapidly, he replies. "I see them now on all the mountains, even Artificial Intelligence Mountain. All of them begin at User Mountain, the largest one. Then they split up, some go straight to Programming Mountain, others go to the Education Range. That set of mountains includes Computer Education Mountain, Administrative Uses Mountain, Instructional Management Mountain, and Educational Software Mountain. Quite a few of them, have recently ventured up Telecommunications Mountain."

As he begins to turn to leave us, he laughingly says, " Your second question shows your ignorance of our land. There is no one price. The cost of using the facilities of each hostel depends on which mountain you are climbing. Some are more expensive than others. Some mountains have more hostels because it takes more time to gain the skills at each level. You need to visit that old castle. It is our only historical structure, and houses our only information center. All your questions can be answered there. Best of luck to you, in our land; you will need it!"

"Thank you," we shout, as he resumes his rapid pace, and disappears around the foot of User Mountain.

Realizing this is a mountain all novices must conquer, we use our binoculars to follow the widest path. Halfway up, as the main road narrows, there is litter everywhere. Refocusing, we can

see that the entire landscape is covered with books, journals, magazines, discs, tapes, and manuals. We are puzzled...Why? It dawns on us. The mountainside is steep. The novices, new travelers, must have lightened their loads, by discarding these items.

As we search the mountainside, we spot a brick house set off the road. A hostel? What's in there? It's time to get some answers.

Specific Information

As we climb the steps of the castle, the sign over the entrance door, attracts our attention. Notice: Only those with a purpose may enter. Well, we meet the criteria. As we enter, we are forced to step down onto a moving conveyor belt, and we are soon enclosed in a glass box. A voice from the overhead speaker, tells us to push the green button, when we hear our destination. Most of the names of the places sound foreign: 8087 micro- processor production; synthesizer chip patents; natural languge processing program copyrights; and, Prologue processor patents. This must be techno-babble. Finally, we hear a familiar phrase, general information and reservations. As soon as the buttom is touched, the voice says, "Second exit to your right," and the glass box is removed. After our rapid exit from the conveyor belt, we are happy to see a carpeted stairway leading to a set of doors. Outside the doors, a notice is

posted: No checks, only credit cards accepted for all reservations. We open the door, and step into a small office, with only one desk, completely filled with telephones, computers, and printers. As we walk up to the desk, looking for some help, the loud speaker voice gives us instructions, "If you are making reservations, insert your credit card in the Hewett Packard Computer and follow the directions on the screen."

"If you are seeking information, use the IBM Computer. Type in your complete name, last name first, a space, and then your first name. Then follow the directions on the screen."

We enter Brenda's name into the IBM and wait. A message appears on the screen " Are you finished?" "If so, please hit the key marked return." Our first interaction with a computer, and we messed it up. Now they know we are novices.

The computer presents us with one personal question after another. It seems to want to know everything. The question, What is your profession?, is followed by a list of questions that the machine will answer. It's our turn now.

We choose our first question, "How do I learn about computers?", and type the question number. The lengthy machine response is summarized below:

> The hostels on the different mountainsides are teaching centers, complete with computer labs, audio-visual labs, libraries, meeting rooms, classrooms, and cafeterias. The services offered at the different hostels range from college-level beginning user

courses, to telecommunication with electrical engineers in Japan. The prices range from $5.00 for a library card, which is valid in any hostel, to $20.00 per minute for communicating with Japan. The amount of time required ranges from minutes (depending on the task) to 40 hours per week for a 4 week beginning user course. All courses use a pre-post test format and all services have a money-back guarantee.

We agree that this land has plenty to offer, if you have the time and the money. We wanted more specific information about what the special educators were doing here.

The list of questions had reappeared on the screen. We selected our second question, " What problems can I expect with computers?" A sense of self-preservation caused an immediate response: "Few computer problems; many computer user problems".

Another menu (i.e., list of choices) appears. The title reads, Problems in which area? , followed by a listing of all mountains, including the ones our little gremlin had named. We selected User, typed the number, and a message appeared:

" Since you are a special educator, all problems listed are the results of a survey of the user group in your profession. Problems in this land are always prefaced by definitions, and a message about the current state of the art in your chosen area."

"Tap the S key to stop, and restart the printing on your screen."

User Problems

User: Anyone who interacts with a computer to complete a task.

State of the Art: Users of today benefit from the results of many years of labor. They have quality computers, a wide variety of software, quality accessories, carefully documented user manuals, a wide variety of instructional materials for learning about computers, and knowledgeable individuals in their communities, who can provide assistance. Most of the new machines can be connected with devices which can generate or respond to electronic signals, like light pens, videodisc players, scientific instruments, and electronic musical instruments. Many accessories have been designed which allow physically handicapped individuals to communicate, using computers. There are many accessories, which enable blind individuals to use computers.

Problems (Majority Opinion):

1. There is too much available information resulting in information overload (Shane, 1983).

2. Typing skills are a prerequisite, if the full power of computers is to be realized.

3. Readers will encounter conflicting information, such as don't learn BASIC, it is too difficult (Judd, 1983) versus BASIC is easy to learn (Machrone, 1983).

4. Finding reliable sources of information is difficult. Reasoning is needed to sort out facts from propaganda and advertising (Shane, 1983).

5. Parent and societal pressure may cause schools to acquire computers rapidly (Judd, 1983), while experienced users suggest planning in detail before purchasing (Railsback, 1983).

6. New products and systems are created and marketed daily, with little standardization in the industry (Walker, 1983).

7. New software created for new machines, will probably not work on old machines. Old software and accessories may not work with the new machines.

8. New terms and buzzwords are created, as needed. There are no standard definitions. For example, the term compatible has now been subdivided into four levels of compatibility (Sandler, 1983).

9. Computers are costly items. Having access to a computer, when it is needed, is difficult. Having enough computers for students and teachers requires good management and planning.

10. Even when computers are donated to the schools, budgets must include maintenance and repair costs.

11. In special education classrooms, the cost of accessories, and software, may surpass the initial cost of the computer within two years.

12. Many hours are needed to understand how computers work, and to comprehend the structure and design of software programs (i.e., the instructions for the computer to follow).

13. Courses designed to introduce novices to computers, may teach you skills you don't really need, and leave out skills you would like to learn.

14. Courses designed for businesses, secretaries, or clerical personnel, will probably not meet your needs.

15. Inservice courses within school districts, will continually change. As new computers and software packages are introduced, the skills of current users must be updated.

16. Instructors who have been teaching computer courses for six or more years, may be burned out.

17. Finding knowledgeable users, with spare time to assist novices, may be difficult.

18. Once you have learned about a computer, and software, it is imperative that you keep reading and learning, due to rapidly changing technology.

19. Teachers may find themselves engaged in fund-raising activities to purchase equipment or software.

20. Parents, whose children have computers at home, may bring pressure to bear on classroom teachers.

21. Trusting a machine is very difficult. Previous experiences with automobiles, movie projectors, and video-recorders can result in fear of mechanical devices.

22. Most novice users are afraid that they will damage the expensive machine. Computers do get tired, and need new parts. But, most damage is caused by spilling liquids on a keyboard, or by repeated power surges, which occur when there are electital

problems in the building.

If all these problems occur going up the first mountain, no wonder the sun shines only at the top. To go on, would require a lot of perseverance, or strong motivation. We decided to learn more. Since we knew, now, that software was needed to run the computer, we chose that important area when the menu reappeared. Again, we were informed that the information was from special educators, and the format of the information was the same.

Educational Software

Educational Software: Programs designed to provide computer-assisted instruction(CAI) for individuals. Note-- the computer does not teach, but helps with instructional tasks.

State of the art: Research suggests that CAI is useful with slow learners, as well as an effective way for all students to learn (Wheatley, 1983). Intelligent tutoring programs, which emulate some teacher behaviors, are available for some subjects. A majority of these programs were developed by researchers in artificial intelligence (AI) (Sleeman & Brown, 1982). Simulations allow students to explore worlds, which are too dangerous, too complex, or too fragile for actual study in the classroom. These programs allow the user to manipulate factors in the environment and control the results. The majority of available educational software is drill and practice, which emphasizes low-level skills, such as basic arithmetic computation. There is a wealth

of software catalogs, magazines which review new software, and an on-line (i.e., computer-based) software evaluation system, RICE. There are also companies, which allow you to test or evaluate programs. Some companies have special equipment, which allows users to down-load (i.e., copy into your computer memory), programs from their computer.

Problems:

1. The greatest obstacle to using educational software is the complex and time-consuming process of finding and evaluating educational programs.

2. Many of the products on the market have never been field-tested (Hofmeister, 1984).

3. Although there are numerous sets of criteria and guidelines for evaluation, some software is unavailable for preview (Hunter, Dearborn, & Snyder, 1983).

4. Tutorials and simulations, the most valuable for handicapped learners, are the most difficult to produce and will come along slowly (Williams, 1983).

5. Many products which have been field-tested, still have major errors in the program. These buys will cause the program to stop in the middle of a work session, or allow user's to win games without ever touching the keyboard.

6. Many programs, termed "fluffware"(Victor, 1983), use a lot of graphics (i.e., computer pictures) and sound, but have no educational content or value.

7. There is a need for programs which collect and store

information on interactions with the learner.

8. There is a shortage of programs which allow teachers to modify the content (e.g., the vocabulary is too difficult for the students).

9. There is and always will be, a need for programs which give alternative responses to students input (i.e., programs that are flexibile). Programs which make decisions based on input, will continue to be developed by those interested in AI.

10. The cost of evaluating software, including the purchase price of magazines and catalogs, cost of mailing, and cost in terms of man-hours is extremely high.

11. Locating and visiting software libraries, for learning to use the packages, and then evaluating them, requires time.

12. The users frustration level increases with each software program that they find unsuitable for their needs.

13. Some users tend to take what software they can find, and then use it for a variety of purposes. Many of these purposes are not educationally sound.

14. Even when software is found that is usable, teachers must use management skills to ensure that all learners, who benefit, have time on the computer.

15. Programmers are not accountable for the results of their product. Therefore, teachers must monitor the ethical and moral values inherent in the software. For example, should emotionally disturbed youngsters be repeatedly exposed to software which uses violence as a reinforcer for correct answers (i.e., video-game

format software)?

16. Due to copyright laws, users may be allowed to make only one back-up copy of a program. Some companies provide back-ups by mail, at purchasers request. Young students, or aggressive students may damage discs. This legal problem is yet to be resolved.

17. Users who want to use telecommunications for either software evaluation, or previewing of programs, need to learn many terms, purchase modems and programs for communication, and pay for the cost of the telephone time.

We now understood why some special educators headed straight to Programming Mountain. If they used educational software on their climb to User Mountain, they probably encountered many of these problems. We had no problem choosing our next area of interest.

Programming

Programming: The act of giving the computer a sequentially-organized, precise set of directions for completing a task, in a language it understands.

State of the Art: All microcomputers understand the language of BASIC. Programs have been written which allow computers to follow directions written in C, COBOL, FORTH, FORTRAN, LISP, LOGO, PASCAL,, PILOT, PROLOG, and SMALLTALK. Programs are available, which allow the computer to follow the instructions (i.e., run

the program) very rapidly (i.e., compilers). Others programs have special tools, which assist programmers in their work (i.e., editors, mapping functions, error monitors). Teachers interested in playing with graphics, and music, or writing simple multiple-choice, yes/no type programs, can use authoring languges, such as Apple Super-Pilot. Programming can be learned, independently or by taking a course. Resources, such as books, audio-visual materials, and magazines, are readily available. If you enjoy being creative, helping others by making things, and learning, programming can be fun (Brooks, 1975). Programming forces you to think logically, and may result in better observational, and problem-solving skills.

Problems:

1. Although many special educators are skilled at task analysis, expressing those steps in a computer language is difficult.

2. Finding a suitable tutorial book, or course with quality instruction requires time, and self-analysis. Learning to program, without feedback from knowledgeable individuals, can be a frustrating, and time-consuming process.

3. Finding a computer to use, either in your home, or close to work or home requires planning and sometimes administrative cooperation. It is not recommended that novices purchase their own computers; there are many sitting idle in computer laboratories, and schools.

4. Finding spare time to learn the vocabulary and rules

(i.e., syntax) of the language, writing programs, typing programs into the computer, finding errors in the program, and then improving it, is a major problem for teachers.

5. Learning the language of programmers, such as the difference between a command and a function, requires additional memory.

6. There is no one right way to write a program. This fact encourages creativity, and fosters frustration.

7. There are programming tricks, which are like frosting on the cake. Finding a source of knowledge, willing to share trade secrets, requires establishing trust, or paying for such information, by taking computer courses.

8. Each programming language was developed to serve a particular purpose. Finding the one, or ones that will meet your needs, is like searching for a book that will teach you everything you need to know about computers.

9. New languages or improved versions of languages are marketed weekly. Each language has a different set of commands, different functions, and will work on only specific computers.

10. Soliciting an unbiased opinion of which language is best for your needs, is impossible because all programmers favor the language they use.

11. The languages of AI, which process words best, and are excellent for developing simulations and tutorials, require computers with a lot of memory, and a good instructor.

12. If you do not program frequently, you will forget the

commands in a short time.

13. If you do program frequently, something else in your life will suffer, as this task is best accomplished in solitude, especially debugging.

14. If you are compulsive, beware of superstitutious behaviors, such as walking around the machine three time, or talking to it, which indicate that it is time to quit for a while and regroup.

15. Forgetting to take breaks, can result in eyestrain, neck and back problems. Poor posture, or the noise from printers will cause excessive irritability.

16. Deciding what to do with your completed masterpiece (i.e., your program), brings about a completely new set of problems (i.e., to publish or not to publish).

17. Addiction to programming is time-consuming, expensive, and results in a limited social life. Awareness of addiction requires careful self-analysis of behaviors. It is usually discovered by concerned others, if they are still around.

We concluded that novices who have a curriculum activity they want put on a disc, might be better off to hire a programmer to write it. Weary, and beginning to wonder if this is worth all the trouble, we decided to find out how things are in the area of Computer Education. That's why we came here in the first place.

Computer Education

Computer Education: Teaching students to access and use the vast amounts of information that will exist in the future. The development of all skill and attitudes needed for this task.

State of the Art: The exact composition of a computer education program and the sequence for instruction has not yet been determined. It is apparent that students need to know about the hardware (i.e., computer, and accessories), how to use the computer for solving a problems or completing a task (i.e., keyboarding skills, running applications programs), be familiar with the logic of programming, and be aware of history and social issues which are related to computers. Scope and sequence curriculum materials are being developed in this area. Programs and games are advertised, which teach keyboarding, or typing skills. Special programs designed for children are surfacing daily. Both children and adults like working with computers, as they receive instant feedback on their responses (i.e., input), and they have control (i.e., power) that is not found in most interactive situations.

Problems:

1. Adapting the computer for use by students who cannot read, or type is an expensive, time-consuming task.

2. Managing the students time, so that everyone has an

opportunity to use the computer, requires skill.

3. Smart systems that talk, and respond to speech input, have limited vocabularies, and some require special installation procedures. Some cause failure of software programs designed to function without such accessories.

4. Selecting software for students to use, is difficult. For example, there are at least 10 different versions of LOGO. How do they differ, and do they run on your available machine?

5. In order for teachers to educate the students, they must first have the necessary skills, attitudes, and time for instruction.

6. Teachers may find that students differ radically in their ability to interact easily with computers. Students will exhibit less fear of mistakes on the machine than the teachers. Some students will learn to program, and advance more rapidly than the teacher. Students with computers in their homes, may have more initial knowledge than the teacher.

7. Organizing the available tasks for students, in a structured sequence requires basic knowledge of all areas. Should you introduce programming first, which will help them in understanding how the word processing program works, or teach them to type first?

8. The attitude of students towards computers is dependent on software quality, hardware reliability, and the teacher's management skills (Hofmeister, 1984).

9. Research is needed in cognitive development, principles of

learning adapted to computer instruction, and instructional technology (i.e., use of accessories, such as voice input devices, vs. keyboarding).

Quite a responsibility there, we sure have a lot to learn. This sounds more difficult than teaching simple addition. Is there a program available for scheduling students, according to task? Let's look at Instructional Management. Maybe some answers await us there.

Instructional Management

Instructional Management: The systematic control of instruction by the computer, which includes testing, diagnosis, learning prescriptions, and thorough record keeping (Burke, 1982).

State of the Art: Programs are available for the following applications: Assessment and development of IEP's (Bennett, 1982; Hannaford & Sloane, 1981); assessing, monitoring, and evaluating progress in IEP goals (Deno, Mirkin, & Wesson, 1984); Management and retrieval of student records (Brady & Dodge, 1982; Hofmeister, 1984); data-based decision-making for selection of instructional strategies (Hasselbring, & Hamlett, 1982); and, interactive diagnositics and remedial instruction (Bennett, 1982; Sleeman, & Brown, 1982). New programs are being developed and marketed at a rapid pace.

Problems:

1. Teachers are worried that they will be replaced as decision-makers.

2. Although all the programs have been designed for individualized instruction, few of them consider or suggest instructional methods.

3. Due to the focus on accountability, teachers are fearful that such systems will be forced upon them.

4. It takes time to transfer data and information from written records to computer, in addition to the time needed for learning the system.

5. Trust in the computer is a prerequisite. Your student data is now in the memory of a computer, or being stored on a disc.

6. If the computer is to be used daily to monitor individual progress, how many computers are needed per room?

7. If data or information from the program, is to be given to parents, or others, a printer is needed.

8. Security measures must be considered, because by law, only certain individuals have access to student records.

9. The programs and systems range in cost from hundreds of dollars to thousands of dollars.

It is apparent that this is the strongest area thus far, with the fewest problems. But where were the systems for scheduling, and word processing. They must be in Administrative Uses.

Administrative Uses

Administrative Uses: Tasks which are part of the professional, business, or managerial aspects of education, such as correspondence, research, budgeting, scheduling, and record-keeping.

State of the Art: In special education, there are computer systems for the following: tracking and managing information on students referred for evaluation; producing required federal and state reports; finding information on materials, instructional programming, and developing IEP's (Brown, 1982). Most of these information management systems are designed for mainframe computers(i.e., big room-size machines). Teachers using microcomputers, can develop individual management systems using software packages developed for business, or can use telecommunication to talk to the mainframes. The micro packages, called applications programs, can be used for these tasks: word-processing (i.e., composing and printing written documents); data-based management (i.e., storing, retrieving, and printing graphs or reports from information stored in files); and, financial record-keeping (i.e., storing numerical data, manipulating that data, and printing reports). New products are copyrighted daily, as programmers improve and/or extend

existing packages. Educators who conduct research, develop their own forms, keep records, manage budgets, or write letters or reports, will find these packages to be invaluable time-savers.

Problems:

1. The packages range in price from free to $500, and price is not indicative of ease of use or features included.

2. Due to the number of packages, and their purposes, finding a package to meet your needs, is a time-consuming process.

3. A majority of the manuals which come with the packages, are unreadable. Purchasers take expensive courses to learn to use their expensive packages. A new publication, Documentation Etc., is directed towards teaching publishers how to write manuals.

4. Some packages require large amounts of computer memory, and most packages are useless without a printer.

5. The packages can be used by students, but teachers must first master all the commands. This can be a lengthy and confusing process.

6. Having access to a computer is a must. Writing first drafts on paper and then using the computer, defeats the purpose of the package.

7. Many of the packages are marketed initially with errors, or limitations. In many cases, logic and problem-solving skills are a necessity.

Similar problems seem to exist in all these areas. Maybe,

our final area, where teachers have recently been seen, telecommunications, will be different.

Telecommunications

Telecommunications: Using telephone lines to send messages, data, or text from one computer to another.

State of the Art: Computers with the proper hardware and software, can be used for reading messages on bulletin boards, putting messages on these boards, accessing data-bases with many types of information, making reservations, finding jobs, downloading software programs (i.e., putting a program into the memory of your machine, or to disc, from another computer), communicating directly with other professionals, holding conferences with many professionals at the same time, and using powerful statistical and applications packages of mainframe computers. New hardware, accessories, and programs used for communication between computers, are being marketed daily. Telecommunications can be local or remote. That is, a mainframe computer in a building may communicate with micros in the same building, using telephone lines and special equipment. Computer users in one part of the world can also talk to users in other parts of the world. Telecommunications is the fastest way of sending mail and searching for and retrieving needed information for research or personal needs. Many professionals and

businessmen do their paperwork at home or on the road, and use the telephone to commute to work.

Problems:

1. The equipment and software needed for telecommunications is expensive.

2. One piece of equipment and/or software will not allow you to communicate with all other computers. Each computer differs in the rate at which it transmits signals, and the way it transmits and receives.

3. There are many rules, procedures and terms, which must be mastered. Defler & Stallings have defined ninety-nine terms, you will encounter (1983).

4. Your telephone bill will increase, and some companies are charging users extra fees for using telephone lines to communicate.

5. Some bulletin boards are free, but most services require an initial membership fee. They bill users by the minute, for time they spend communicating with the system.

6. The cheapest rates for these services are from 5:00 P.M. through 8:00 A.M., times when teachers are not in the schools.

7. Downloading information requires either a lot of computer memory, or many discs.

8. The information which is downloaded, may need to be modified before it can be used with applications packages, such as word-processors.

9. Many complex technical problems need to be resolved,

before all computers can talk to all other computers. At this time, computers manufactured by the same company cannot communicate with each other.

Now we understood why the special educators visitied this mountain last, it was the most expensive one. We could see that we had a lot of learning ahead of us.

On our way out of the castle, we began discussing how we could best cope with the challenges, and problems. We kept wondering, if our glass enclosure was recording our conversation. As we walked down the path, to the door we had entered, long ago, we decided that we would not reenter this land until we were prepared. We needed a plan for action.

Developing Self-Awareness

Back in the sanctity of our cubby-hole of an office, we began to list the constraints that were present in all areas. It seemed that the factors of time, money, too many choices, inferior products, and undeveloped products would hamper our progress. We decided to visit a computer shop, and see what these machines were now capable of doing.

We played games, used some arithmetic drill and practice programs, wrote a letter on a word processor, and watched a demonstration of a data-base management system. As we played the games, we began to feel more comfortable with the touch-sensitive keyboard. When our letters were safely out of the computer, and

printed on paper, we relaxed a little. We didn't really trust these machines yet, but the others in the shop seemed pretty calm. In two and a half hours, no one seemed to have any serious accidents with their machines. Maybe the reminders on the top of the screen helped everyone remember those word-processing commands. When the salesman did the demonstration for us, he used something called a help menu, which listed commands that he had forgotten. Maybe these computers weren't friendly, but the programs offered some help.

We stopped at a nearby bookstore, and visited two sections, the computer books, and the magazines. We left quickly, without purchasing a single item. Most of the books, and magazines, seemed dedicated to users of specific computers. We didn't even know what computer we were getting for our classroom, yet.

We had taken the first step, we decided we wanted to learn about computers. We had already thought of numerous uses for that word-processor in our classroom. Two computers were going to be ours. One of our parents, an accountant, had offered us his old computer, and we would get one from the funds raised by the P.T.A. We decided to find out the brands, and models, so we wouldn't waste any time learning about other computers.

After two phone calls, we were ready to begin our plan. First, we listed all the factors in our lives that would either assist or hamper our efforts. We then considered different learning environments, and their factors. Finally, we listed factors, which resulted from a combination of our personal

characteristics and those of the environment. For example, both of us felt that we needed feedback on our progress, which meant that we would need to find a course to take. We knew that many questions would arise, as we learned terms and concepts about electronic machines. So, we needed a friendly, responsive instructor. One of us was assertive enough to ask questions in class, when concepts or examples were not clear; even when the question seemed ridiculous. The other would not. This factor would effect our environment and our progress.

We made an outline of our factors, grouping them into our three major categories: Personal: Situational; and Personal-Situational. Next we reviewed the six computer areas and made some tentative decisions. We knew we would have to become Users first, and we needed something to work on during that process. We decided to learn to program. We suspected that knowledge of programming, would help us in evaluating software. We used time as a framework for setting goals. Later, we realized that we should have used money.

We needed a course that would teach us computer terminology, and programming. We gave ourselves six months to accomplish our long term goals. Fall would be here, and beginning of the year activities would limit our time.

As we began looking for a course, we added other factors to our outline. We found courses everywhere, teaching different languages, on different computers. We found differences in computer lab facilities, hours of operation, and friendliness of

the lab assistants. We also found human resources to be the most valuable part of our search. We talked to students who had taken courses, instructors of the courses, media specialists who worked with computers, and programmers, who knew the languages. These human resources saved us time, and money.

Talking with them convinced us that not only did we need long-term goals, but we better have some short-term goals, if we were to keep a high level of motivation. Our students proved to be an excellent source of motivation. Teaching our students about computers, would keep those terms, concepts, and skills, alive and fresh. For the programming part of the course, we each had a program in mind, that our students could use.

By this time, our outline had grown to two pages. It contained all the constraining and facilitating elements, we considered important (See Table 1, Major Factors in Decision Making and Table 2, Other Major Factors in Decision Making) From this, we devised a chart, which was used to rate factors which were important to us, individually. Table 3, Sample Rating of Factors for Programming, illustrates use of the chart. Our priorities were determined by using a rating scale of 1 to 10. With the aid of the chart, it was easy to see which areas were the most important, and where problems could arise.

We found a course, that had the fewest constraining elements, and enrolled. Shortly, we encountered another stumbling block. Because the computer field changes so rapidly, the textbook for the course was not up-to-date. This meant

Table 1

Major Factors in Decision Making

Factors	Factors
Personal Assessment	
A. Assets 1. Personal Abilities 2. Professional Abilities 3. Personal Characteristics B. Liabilities 1. Personal Commitments (Time, Money, & Energy) 2. Professional Committments (Time, Money, & Energy) 3. Learning Style 4. Preferred Learning Mode 5. Preferred Environment	C. Goal Determination 1. Required for Certification 2. Peer-pressured Motives 3. Personal Motives 4. Professional Motives D. Strength of Motivation 1. Long-term Goals 2. Short-term Goals 3. Reinforcement (a) Schedule (b) Reinforcers (c) Time-out
Situational Assessment	
A. Classroom Instruction 1. Instructor (Attitude, Role Model, Accessibility) 2. Cost (Course, Textbook, Supplies) 3. Time (Class, Homework, Computer, Reading) 4. Content (a) Computer Essentials (b) Terms & Jargon (c) Skills 5. Setting (a) Location (b) Computer Lab (Hours,Staff, Availability) (c) Computers (Type, Model) (d) Language (Basic, Pilot, Logo, Others)	B. Self-Instruction 1. Cost 2. Time Required 3. Need for Feedback 4. Learning Environment 5. Computer Access 6. Method (a)Programmed Text (b)Magazines (c)Media Packages

Table 2

Other Major Factors in Decision-Making

Factors	Factors
Personal- Situational Assessment	
A. Computer Experience	D. Learning Interference
1. Previous Experience	1. Distractibility
2. Fear of Machines	2. Self-consciousness
3. Typing Ability	E. Other Needs
B. Personal Assertiveness (Requests Assistance)	1. Competition
1. In Group	2. Structure
2. Individually	3. Solitude
C. Self-awareness	
1. Setting Objectives	
2. Monitoring (Frustration Tolerance, Compulsiveness vs. Persistence, Patience)	
3. Evaluation	
4. Reinforcement	
5. Reflection	
6. Renewal	

Table 3

Sample Rating Of Factors For Programming

ASSETS

Personal Abilities		Professional Abilities		Personal Characteristics	
Creative Writ.	2	Task Analysis	5	Persistence	5
Algebra	5	ABA	5	Patience	7

LIABILITIES

Personal Commitments		Professional Commitments		Learning Style		Preferred Learning Mode		Preferred Environment	
Family	3	Paperwork	5	Serialist	5	Visual	5	Quiet	4
Sports	5	Prof.Meetings	3					Cool	2
Money	5								

GOAL DETERMINATION

Required for Certification		Peer pressured Motives		Personal Motives		Professional Motives	
No	6	Slight	3	Strong	7	Software	5

STRENGTH OF MOTIVATION

Long-term Goals		Short-term Goals		Reinforcement	
Program.=BASIC	8	Stud. Prog.	7	Light Sched.	6
				Avail.Reinf.	7
				Time-out	6

(Table 3 continued)

Sample Rating Of Factors For Programming

CLASSROOM INSTRUCTION

Instructor		Cost		Time	
Attitude	6	Course=$400	2	Class=3Hr.	4
Role Model	7	Text=$25	2	Comp.=6Hr.	2
Tricks	7	Supplies=$30?	2	Read.=6Hr.	2

Content		Setting		Setting	
Comp.Essent.	8	2Miles Away	8	Long Hours	7
BASIC	8	20Computers	6	Friendly Staff	7
		10Printers	6	IBM Comp.	9

SELF INSTRUCTION

Cost		Time Required		Need for Feedback		Learning Environment	
Book=$30	3	Unknown	2	Strong	1	Quiet	8
Magazines=?	1						

Computer Access		Method			
FromSchool	2	Programmed	2		
		Typing Progr.In	1		

finding more current information about our computers. Magazines were within our budget, so we began searching for information. Many of them were too technical for us. We decided it was time to begin monitoring our purchases. Two criteria were set, content of the articles for our purposes, and readibility level. If we could not comprehend the text of randomly selected paragraphs in at least three articles, it went back on the shelf. Later, we found that the computer lab had magazines, we could have checked out. Too late, we had already started our own collection. In the library, we read educational journals, focusing on computers, without encountering this problem. We read and we read and we read.

When we began suffering from information overload, we took time to look back. Finding the first magazine article, which had been totally confusing at first, now made perfect sense. We had learned, we had grown, and we patted ourselves on the back. We began rewarding our good behavior, especially when we began programming. We took time away from computers, and literature on computers. But only, when we had achieved some objective, we had set. Putting the finishing touches on a lengthy grading program, was dutifully rewarded.

We learned a lot about ourselves, and computers in the process. Family and friends accused us of carrying on an affair with a machine. Our dreams consisted of dramas played out inside a computer lab; we began to worry. Doubts began to plague us; maybe we weren't cut out for this programming stuff.

We began looking for small success: " Good, that part of the program works!" We talked to ourselves, and to the computer, "Everything is going to work today, right!" The day came when the program for the students was working correctly. It was finished. As we walked out of the building, a printed copy in hand, the sun was shining, the birds were singing, and the grass was green once again. The cloud of gloom that had hung overhead was gone. The machine had finally learned; it had a new master.

The Future

This was just the beginning. The confidence was there. The mountains of the land of computers appeared smaller. The valleys, where knowledge could be shared with others, made it all worthwhile. Eagerly new mountains were challenged, the climb to the top was less strenuous, less tiring.

But we had forgotten one thing, in our eagerness, in our compulsion to learn it all: we had started to learn 15 years after it all began. We were playing catch-up. The following occurred, and in all likelihood will happen to you, too:

One day the cloud came back, but this time it was a dense fog. I was overwhelmed. There's just too much knowledge out there, I can't learn it all. I'm tired of computers, reading about computers, and learning about computers. That keyboard can just gather dust, and those internal metal connectors can rust. Who cares anymore?

Was I burned out already? No, it was time for more self-monitoring and some soul-searching.

I had forgotten to look back, to see how far I had come. I had been searching for answers, but had found more questions. My idealism was gone, realism had bloomed in its place. I had truly learned, but couldn't cope with reality. The computer had not caused the problems in our society, and it could not solve them. Man, in quest of fame, fortune, and power was the source. In the race for the high technology stake, ideals had been tossed aside, and, values compromised. The gremlin had given us an inkling of what was to come, but his warning had been ignored. What can you do?

Know yourself, your values, beliefs, weaknesses and strengths. Set your goals accordingly, and you will be prepared to master these machines.

The computer is only a tool. Use it wisely. Share your knowledge with others. It will serve you well.

References

Bennett, R.E. (1982) Applications of microcomputer technology to special education. Exceptional Children, 49(2), 106-113.

Brady, R.C., & Dodge, B.J. (1982) Increasing productivity with microcomputers: Key to improvement of special education in the 1980's. Teacher Education and Special Education, 5(3), 30-35.

Brooks, F.P. (1975) The Mythical Man-Month: Essays on Software Engineering. Reading, MA.: Addison-Wesley.

Brooks, R.A. (1983) Find-path for a PUMA-class robot. Proceedings of the National Conference on Artificial Intelligence. pp.40-44.

Brown, N.P. (1982) CAMEO: Computer-assisted management of educational objectives. Exceptional Children, 49(2), 151-153.

Burke, R.L. (1982) CAI Sourcebook. Englewood Cliffs, N.J.: Prentice-Hall.

Deno, S.L., Mirkin, P.K., & Wesson, C. (1984) How to write effective data-based IEPs. Teaching Exceptional Children. 16(2), 99-104.

Derfler, F.J., & Stallings, W. (1983) Local networks: A guide for the perplexed. PC Magazine, 2(3), 237-268.

Feigenbaum, E.A.,& McCorduck, P. (1983) The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World. Reading, MA.: Addison-Wesley.

Hannaford, A., & Sloane, E. (1981) Microcomputers: Powerful

learning tools with proper programming. Teaching Exceptional Children, 14, 54-57.

Harvey, B. (1983) Stop saying computer literarcy. The Computing Teacher, 10(8), 56-57.

Hasselbring, T.S., & Hamlett, C.L. (1982) Using the microcomputer to apply data-based decision rules for assisting in the selection of instructional strategies for the handicapped. Proceedings of the 20th Annual Convention of the Association for Educational Data Systems. pp. 43-46.

Hofmeister, A. (1984) Microcomputer Applications in the Classroom. New York: Holt, Rinehart, & Winston.

Hunter, B., Dearborn, D., & Snyder, B. (1983) Computer literacy in the K-8 curriculum. Kappan, 65(2), 115-118.

Judd, W. (1983) A teacher's place in the computer curriculum. Kappan, 65(2), 120-122.

Lenat, D.B., & Brown, J.S. (1983) Why AM and Eurisko appear to work. Proceedings of the National Conference on Artificial Intelligence. pp. 236-240.

Lopez, S. (1983) Brave new worlds. PC Magazine, 2(3),419-422.

Machrone, B. (1983) Micro-linguistics: Languages for the PC. PC Magazine, 2(4), 110-156.

Railsback, C.E. (1983) Microcomputers: Solutions in search of problems? Kappan, 65(2), 118-119.

Sandler, C. (1983) The prognosticators pronounce: Future compilations on the PC. PC Magazine, 2(7), 248-257.

Shane, R.L. (1983) The silicon age II: Living and learning in an information epoch. Kappan, 65(2), 126-129.

Sleeman, D., & Brown, J.S. (1982) (Eds.) Intelligent Tutoring Systems. New York: Academic Press.

Stallard, C.K. (1982) Computers and education for exceptional children: Emerging applications. Exceptional Children, 49(2), 102-104.

Toffler, A. (1980) The Third Wave. New York: William Morrow Inc.

Toffler, A. (1983) Previews & Premises. New York: William Morrow Inc.

Victor, J. (1983) Input/Output: Where there's swill there's no way. Creative Computing, 9(10), 10.

Walker, D.F. (1983) Reflections on the educational potential and limitations of microcomputers. Kappan, 65(2), 103-107.

Wheatley,G. (1983) Designing a computer education program for your school. Educational Computer Magazine, 3(5), 51-54.

Williams, L.V. (1983) The California challenge. P.C. World, December, 324-329.